Internet Engineering Task Force (IETF)
Request for Comments: 7341
Category: Standards Track
ISSN: 2070-1721

Q. Sun
Y. Cui
Tsinghua University
M. Siodelski
ISC
S. Krishnan
Ericsson
I. Farrer
Deutsche Telekom AG

August 2014

DHCPv4-over-DHCPv6 (DHCP 4o6) Transport

Abstract

IPv4 connectivity is still needed as networks migrate towards IPv6. Users require IPv4 configuration even if the uplink to their service provider supports IPv6 only. This document describes a mechanism for obtaining IPv4 configuration information dynamically in IPv6 networks by carrying DHCPv4 messages over DHCPv6 transport. Two new DHCPv6 messages and two new DHCPv6 options are defined for this purpose.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc7341.

Table of Contents

1. Introduction

As the migration towards IPv6 continues, IPv6-only networks will become more prevalent. In such networks, IPv4 connectivity will continue to be provided as a service over IPv6-only networks. In addition to provisioning IPv4 addresses for clients of this service, other IPv4 configuration parameters may also be needed (e.g., addresses of IPv4-only services).

This document describes a transport mechanism to carry DHCPv4 messages using the DHCPv6 protocol for the dynamic provisioning of IPv4 addresses and other DHCPv4 specific configuration parameters across IPv6-only networks. It leverages the existing DHCPv4 infrastructure, e.g., failover, DNS updates, DHCP Leasequery, etc.

When IPv6 multicast is used to transport DHCP 4o6 messages, another benefit is that the operator can gain information about the underlying IPv6 network to which the DHCP 4o6 client is connected from the DHCPv6 relay agents through which the request has passed.

2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

3. Terminology

This document makes use of the following terms:

CPE:
: Customer Premises Equipment (also known as Customer Provided Equipment), which provides access for devices connected to a Local Area Network (LAN), typically at the customer’s site/home, to the Internet Service Provider’s (ISP’s) network.

DHCP 4o6 client (or client):
: A DHCP client supporting both the DHCPv6 protocol [RFC3315] as well as the DHCPv4 over DHCPv6 protocol described in this document. Such a client is capable of requesting IPv6 configuration using DHCPv6 and IPv4 configuration using DHCPv4 over DHCPv6.

DHCP 4o6 server (or server):
: A DHCP server that is capable of processing DHCPv4 packets encapsulated in the DHCPv4 Message option (defined below).

DHCPv4 over DHCPv6:
A protocol (described in this document) used to carry DHCPv4 messages in the payload of DHCPv6 messages.

4. Applicability

The mechanism described in this document is not universally applicable. This is intended as a special-purpose mechanism that will be implemented on nodes that must obtain IPv4 configuration information using DHCPv4 in specific environments where native DHCPv4 is not available. Such nodes are expected to follow the advice in Section 9; nodes that do not require this functionality are expected not to implement it, or not to enable it by default. This mechanism may be enabled using an administrative control, or it may be enabled automatically in accordance with the needs of some dual-stack transition mechanism such as [LW4OVER6]. Such mechanisms are beyond the scope of this document.

5. Architecture Overview

The architecture described here addresses a typical use case, where a DHCP client's uplink supports IPv6 only and the Service Provider's network supports IPv6 and limited IPv4 services. In this scenario, the client can only use the IPv6 network to access IPv4 services, so IPv4 services must be configured using IPv6 as the underlying network protocol.

Although the purpose of this document is to address the problem of communication between the DHCPv4 client and the DHCPv4 server, the mechanism that it describes does not restrict the transported messages types to DHCPv4 only. As the DHCPv4 message is a special type of BOOTP message, BOOTP messages [RFC951] MAY also be transported using the same mechanism.

DHCP clients may be running on CPE devices, end hosts, or any other device that supports the DHCP-client function. This document uses the CPE as an example for describing the mechanism. This does not preclude any end host, or other device requiring IPv4 configuration, from implementing DHCPv4 over DHCPv6 in the future.

This mechanism works by carrying DHCPv4 messages encapsulated within the newly defined DHCPv6 messages. The DHCPv6-relay encapsulation is used solely to deliver DHCPv4 packets to a DHCPv4-capable server, and does not allocate any IPv6 addresses nor does it provide IPv6-configuration information to the client. Figure 1, below, illustrates one possible deployment architecture of this mechanism.

The DHCP 4o6 client implements a new DHCPv6 message called DHCPv4-query, which carries a DHCPv4 message encapsulated in the new DHCPv4 Message option. The DHCPv6 message can be transmitted either via DHCPv6 Relay Agents or directly to the DHCP 4o6 server.

The server replies with a new DHCPv6 message called DHCPv4-response, which carries the DHCPv4 message from the server, encapsulated in the DHCPv4 Message option.

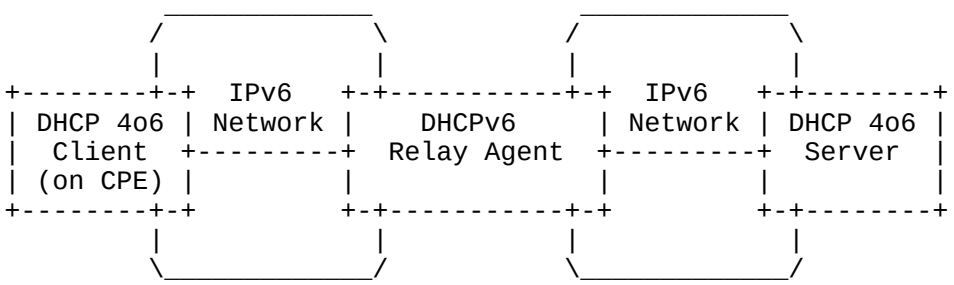

Figure 1: Architecture Overview

Before the client can use DHCPv4 over DHCPv6, it MUST obtain the necessary IPv6 configuration. The client requests the DHCP 4o6 Server Address option from the server by sending the option code in an Option Request option as described in [RFC3315]. If the server responds with the DHCP 4o6 Server Address option, it is an indication to the client to attempt using DHCPv4 over DHCPv6 to obtain IPv4 configuration. Otherwise, the client MUST NOT use DHCPv4 over DHCPv6 to request IPv4 configuration.

The client obtains the address(es) of the DHCP 4o6 server(s) from the DHCP 4o6 Server Address option and uses it (them) to communicate with the DHCP 4o6 servers as described in Section 9. If the DHCP 4o6 Server Address option contains no addresses (is empty), the client uses the well-known All_DHCP_Relay_Agents_and_Servers multicast address to communicate with the DHCP 4o6 server(s).

Before applying for an IPv4 address via a DHCPv4-query message, the client must identify a suitable network interface for the address. Once the request is acknowledged by the server, the client can configure the address and other relevant parameters on this interface. The mechanism for determining a suitable interface is out of the scope of the document.

6. New DHCPv6 Messages

Two new DHCPv6 messages carry DHCPv4 messages between the client and the server using the DHCPv6 protocol: DHCPv4-query and DHCPv4-response. This section describes the structures of these messages.

6.1. Message Types

DHCPV4-QUERY (20): The DHCP 4o6 client sends a DHCPv4-query message to a DHCP 4o6 server. The DHCPv4 Message option carried by this message contains a DHCPv4 message that the DHCP 4o6 client uses to request IPv4 configuration parameters from the server.

DHCPV4-RESPONSE (21): A DHCP 4o6 server sends a DHCPv4-response message to a DHCP 4o6 client. It contains a DHCPv4 Message option carrying a DHCPv4 message in response to a DHCPv4 message received by the server in the DHCPv4 Message option of the DHCPv4-query message.

6.2. Message Formats

Both DHCPv6 messages defined in this document share the following format:

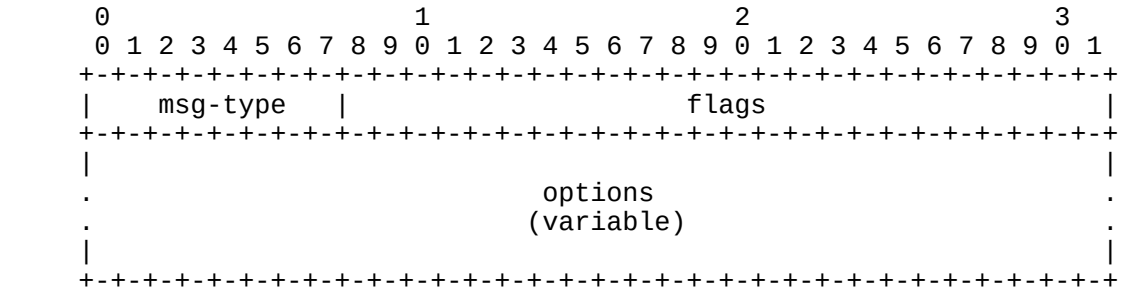

Figure 2: The Format of DHCPv4-query and DHCPv4-response Messages

msg-type: Identifies the message type. It can be either DHCPV4-QUERY (20) or DHCPV4-RESPONSE (21) corresponding to the contained DHCPv4-query or DHCPv4-response, respectively.

flags: Specifies flags providing additional information required by the server to process the DHCPv4 message encapsulated in the DHCPv4-query message, or required by the client to process a DHCPv4 message encapsulated in the DHCPv4-response message.

options: Options carried by the message. The DHCPv4 Message Option (described in Section 7.1) MUST be carried by the message. Only DHCPv6 options for IPv4 configuration may be included in this field. It MUST NOT contain DHCPv6 options related solely to IPv6, or IPv6-only service configuration.

6.3. DHCPv4-query Message Flags

The "flags" field of the DHCPv4-query is used to carry additional information that may be used by the server to process the encapsulated DHCPv4 message. Currently, only one bit of this field is used. Remaining bits are reserved for the future use. The "flags" field has the following format:

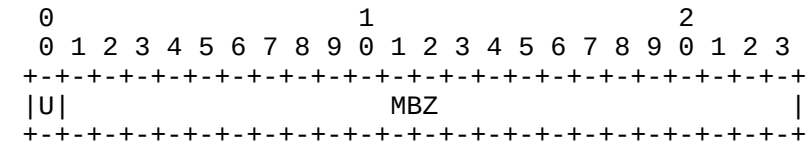

Figure 3: DHCPv4-query Flags Format

U: Unicast flag. If set to 1, it indicates that the DHCPv4 message encapsulated within the DHCPv4-query message would be sent to a unicast address if it were sent using IPv4. If this flag is set to 0, it indicates that the DHCPv4 message would be sent to the broadcast address if it were sent using IPv4. The usage of the flag is described in detail in Section 8.

MBZ: Bits MUST be set to zero when sending and MUST be ignored when receiving.

6.4. DHCPv4-response Message Flags

This document introduces no flags to be carried in the "flags" field of the DHCPv4-response message. They are all reserved for future use. The DHCP 4o6 server MUST set all bits of this field to 0 and the DHCP 4o6 client MUST ignore the content in this field.

7. New DHCPv6 Options

7.1. DHCPv4 Message Option Format

The DHCPv4 Message option carries a DHCPv4 message that is sent by the client or the server. Such messages exclude any IP or UDP headers.

The format of the DHCPv4 Message option is:

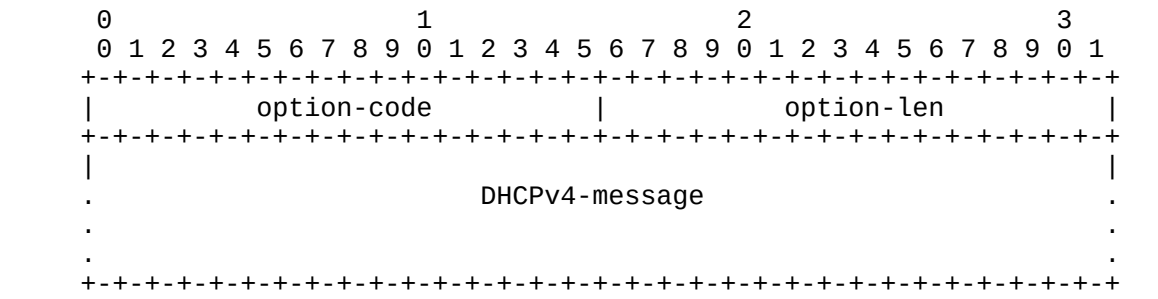

Figure 4: DHCPv4 Message Option Format

option-code: OPTION_DHCPV4_MSG (87).

option-len: Length of the DHCPv4 message.

DHCPv4-message: The DHCPv4 message sent by the client or the server. In a DHCPv4-query message, it contains a DHCPv4 message sent by a client. In a DHCPv4-response message, it contains a DHCPv4 message sent by a server in response to a client.

7.2. DHCP 4o6 Server Address Option Format

The DHCP 4o6 Server Address option is sent by a server to a client requesting IPv6 configuration using DHCPv6 [RFC3315]. It carries a list of DHCP 4o6 servers' IPv6 addresses that the client should contact to obtain IPv4 configuration. This list may include multicast and unicast addresses. The client sends its requests to all unique addresses carried in this option.

This option may also carry no IPv6 addresses, which instructs the client to use the All_DHCP_Relay_Agents_and_Servers multicast address as the destination address.

The presence of this option in the server's response indicates to the client that it should use DHCPv4 over DHCPv6 to obtain IPv4 configuration. If the option is absent, the client MUST NOT enable DHCPv4-over-DHCPv6 functionality.

The format of the DHCP 4o6 Server Address option is:

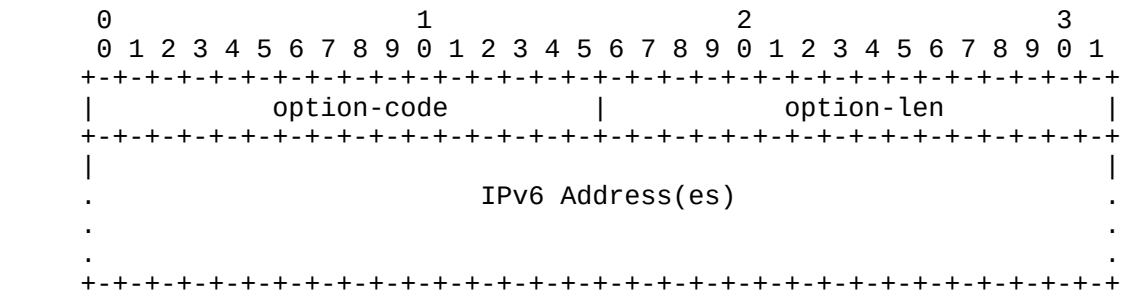

Figure 5: DHCP 4o6 Servers Address Option Format

option-code: OPTION_DHCP4_O_DHCP6_SERVER (88).

option-len: Length of the IPv6 address(es) carried by the option, i.e., multiple of 16 octets. Minimal length of this option is 0.

IPv6 Address: Zero or more IPv6 addresses of the DHCP 4o6 server(s).

8. Use of the DHCPv4-query Unicast Flag

A DHCPv4 client conforming to [RFC2131] may send its DHCPREQUEST message to either a broadcast or unicast address depending on its state. For example, a client in the RENEWING state uses a unicast address to contact the DHCPv4 server to renew its lease. A client in the REBINDING state uses a broadcast address.

In DHCPv4 over DHCPv6, IPv6 is used to deliver DHCPv4 messages to the DHCP 4o6 server. There is no relation between the outer IPv6 address and the inner DHCPv4 message. As a result, the server is unable to determine whether the received DHCPv4 messages should have been sent using broadcast or unicast in IPv4 by checking the IPv6 address.

In order to allow the server to determine the client’s state, the Unicast flag is carried in the DHCPv4-query message. The client MUST set this flag to 1 when the DHCPv4 message would have been sent to the unicast address if using DHCPv4 over IPv4. This flag MUST be set to 0 if the DHCPv4 client would have sent the message to the broadcast address in IPv4. The choice whether a given message should be sent to a broadcast or unicast address is made based on the [RFC2131] and its extensions.

Note: The Unicast flag reflects how the DHCPv4 packet would have been sent; not how the DHCPv6 packet itself is sent.

9. DHCP 4o6 Client Behavior

The client MUST obtain necessary IPv6 configuration from a DHCPv6 server before using DHCPv4 over DHCPv6. The client requests the DHCP 4o6 Server Address option using the Option Request option (ORO) in every Solicit, Request, Renew, Rebind, and Information-request message. If the DHCPv6 server includes the DHCP 4o6 Server Address option in its response, it is an indication that the client can use DHCPv4 over DHCPv6 to obtain the IPv4 configuration (by sending DHCPv4 messages encapsulated in DHCPv4-query messages).

The client MUST NOT use DHCPv4 over DHCPv6 to request IPv4 configuration if the DHCPv6 server does not include the DHCP 4o6 Server Address option. If the IPv6 configuration that contained the DHCP 4o6 Server Address option subsequently expires, or if the renewed IPv6 configuration does not contain the DHCP 4o6 Server Address option, the client MUST stop using DHCPv4 over DHCPv6 to request or renew IPv4 configuration. However, the client continues to request DHCP 4o6 Server Address option in the messages sent to the DHCPv6 server as long as it desires to use DHCPv4 over DHCPv6.

It is possible in a multihomed configuration for there to be more than one DHCPv6 configuration containing a DHCP 4o6 Server Address Option active at the same time. In this case, the configurations are treated as being independent, so that when any such configuration is active, a DHCPv4-over-DHCPv6 function may be enabled for that configuration.

An implementation may also treat such configurations as being exclusive, such that only one is kept active at a time. In this case, the client keeps the same configuration active continuously as long as it is valid. If that configuration becomes invalid but one or more other configurations remain valid, the client activates one of the remaining valid configurations.

Which strategy to follow is dependent on the implementation: keeping multiple configurations active at the same time may provide useful redundancy in some applications but may be needlessly complex in other cases.

If the client receives the DHCP 4o6 Server Address option and DHCPv4 [RFC2131] is used on the interface over which the DHCPv6 option was received, the client MUST stop using the IPv4 configuration received using DHCPv4 on this interface. The client MAY send a DHCPRELEASE to the DHCPv4 server to relinquish an existing lease as described in Section 4.4.6 of [RFC2131]. The client MUST NOT use DHCPv4 on this interface as long as it receives DHCP 4o6 Server Address option in the messages received from the DHCPv6 server.

If the client receives a DHCP 4o6 Server Address option that contains no IP addresses, i.e., the option is empty, the client MUST send its requests to the All_DHCP_Relay_Agents_and_Servers multicast address. If there is a list of IP addresses in the option, the client SHOULD send requests to each unique address carried by the option.

If the client obtained stateless IPv6 configuration by sending an Information-request message to the server, the client MUST follow the rules in [RFC4242] to periodically refresh the DHCPv4-over-DHCPv6 configuration (i.e., list of DHCP 4o6 servers) as well as other configuration data. The client that obtained stateful IPv6 configuration will refresh the status of DHCPv4-over-DHCPv6 function when extending a lifetime of acquired IPv6 address (Renew and Rebind messages).

The client MUST employ an IPv6 address of an appropriate scope from which to source the DHCPv4-query message. When the client sends a DHCPv4-query message to the multicast address, it MUST use a link-local address as the source address as described in [RFC3315]. When the client sends a DHCPv4-query message using unicast, the source address MUST be an address of appropriate scope, acquired in advance.

The client generates a DHCPv4 message and stores it verbatim in the DHCPv4 Message option carried by the DHCPv4-query message. The client MUST put exactly one DHCPv4 Message option into a single DHCPv4-query message. The client MUST NOT request the DHCP 4o6 Server Address option in the DHCPv4-query message.

The client MUST follow the rules defined in Section 8 when setting the Unicast flag based on the DHCPv4 destination.

On receiving a DHCPv4-response message, the client MUST look for the DHCPv4 Message option within this message. If this option is not found, the DHCPv4-response message is discarded. If the DHCPv4 Message option is present, the client extracts the DHCPv4 message it contains and processes it as described in Section 4.4 of [RFC2131].

When dealing with IPv4 configuration, the client MUST follow the normal DHCPv4 retransmission requirements and strategy as specified in Section 4.1 of [RFC2131]. There are no explicit transmission parameters associated with a DHCPv4-query message, as this is governed by the DHCPv4 "state machine" [RFC2131].

The client MUST implement [RFC4361] to ensure that the device correctly identifies itself. It MUST send a ’client identifier’ option when using DHCPv4 over DHCPv6.

10. Relay Agent Behavior

When a DHCPv6 relay agent receives a DHCPv4-query message, it may not recognize this message. The unknown message MUST be forwarded as described in [RFC7283].

A DHCPv6 relay agent that can recognize DHCP 4o6 messages MAY allow the configuration of a separate set of destination addresses for such messages in addition to the destination addresses used for relaying the other DHCPv6 messages. To implement this function, the relay checks the received DHCPv6 message type and forwards according to the following logic:

1. If the message type is DHCPV4-QUERY, the packet is relayed to the configured DHCP 4o6 Server's address(es) in the form of a normal DHCPv6 packet (i.e., DHCPv6/UDP/IPv6).

2. For any other DHCPv6 message type, forward according to section 20 of [RFC3315].

The above logic only allows for separate relay destinations configured on the relay agent closest to the client (single relay hop). Multiple relaying hops are not considered in the case of separate relay destinations.

11. DHCP 4o6 Server Behavior

When the server receives a DHCPv4-query message from a client, it searches for the DHCPv4 Message option. The server discards a packet without this option. In addition, the server MAY notify an administrator about the receipt of this malformed packet. The mechanism for this notification is out of scope for this document.

If the server finds a valid DHCPv4 Message option, it extracts the original DHCPv4 message. Since the DHCPv4 message is encapsulated in the DHCPv6 message, it lacks the information that is typically used by the DHCPv4 server, implementing [RFC2131], to make address-allocation decisions, e.g., giaddr for relayed messages and IPv4 address of the interface that the server is using to communicate with a directly connected client. Therefore, the DHCP 4o6 server allocates addresses according to the policies on local address assignment determined by the server administrator. For example, if the DHCPv4-query message has been sent via a relay, the server MAY use the link-address field of the Relay-forward message as a lookup for the IPv4 subnet from which to assign a DHCPv4 address. If the DHCPv4-query message has been sent from a directly connected client,

the server MAY use the IPv6 source address of the message to determine the appropriate IPv4 subnet to use for DHCPv4 address assignment.

Alternatively, the server may act as a DHCPv4 relay agent and forward the DHCPv4 packet to a "normal" DHCPv4 server. The details of such a solution have not been considered by the working group; describing that solution is out of scope of this document and is left as future work should the need for it arise.

The server SHOULD use the "flags" field of the DHCPv4-query message to create a response (server to client DHCPv4 message). The use of this field is described in detail in Section 8.

When an appropriate DHCPv4 response is created, the server places it in the payload of a DHCPv4 Message option, which it puts into the DHCPv4-response message.

If the DHCPv4-query message was received directly by the server, the DHCPv4-response message MUST be unicast from the interface on which the original message was received.

If the DHCPv4-query message was received in a Relay-forward message, the server creates a Relay-reply message with the DHCPv4-response message in the payload of a Relay Message option, and responds as described in Section 20.3 of [RFC3315].

12. Security Considerations

In this specification, DHCPv4 messages are encapsulated in the newly defined option and messages. This is similar to the handling of the current relay agent messages. In order to bypass firewalls or network authentication gateways, a malicious attacker may leverage this feature to convey other messages using DHCPv6, i.e., use DHCPv6 as a form of encapsulation. However, the potential risk from this is no more severe than that with the current DHCPv4 and DHCPv6 practice.

It is possible for a rogue server to reply with a DHCP 4o6 Server Address option containing duplicated IPv6 addresses, which could cause an amplification attack. To avoid this, the client MUST check if there are duplicate IPv6 addresses in a DHCP 4o6 Server Address option when receiving one. The client MUST ignore any but the first instance of each address.

When considering whether to enable DHCPv4-over-DHCPv6, one important consideration is that when it is enabled, this gives the DHCPv6 server the ability to shut off DHCPv4 traffic, and, consequently, IPv4 traffic, on the interface that is configured to do DHCPv4-over-

DHCPv6. For this reason, DHCPv4-over-DHCPv6 should only be enabled in situations where there is a clear trust relationship that eliminates this concern. For instance, a CPE device can safely enable this on its WAN interface, because it is reasonable to assume that an ISP will not accidentally configure DHCPv4 over DHCPv6 service on that link, and that it will be impractical for an attacker to set up a rogue DHCPv6 server in the ISP's network.

13. IANA Considerations

IANA has allocated two DHCPv6 option codes for use by OPTION_DHCPV4_MSG (87) and OPTION_DHCP4_O_DHCP6_SERVER (88) from the "Option Codes" table. Also, IANA has allocated two DHCPv6 message type codes for the DHCPV4-QUERY (20) and DHCPV4-RESPONSE (21) from the "Message Types" table of the "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)" registry. Both tables can be found at <http://www.iana.org/assignments/dhcpv6-parameters/>.

14. Contributors List

Many thanks to Ted Lemon, Bernie Volz, Tomek Mrugalski, Cong Liu, and Yuchi Chen for their great contributions to the specification.

15. References

15.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2131] Droms, R., "Dynamic Host Configuration Protocol", RFC 2131, March 1997.

[RFC3315] Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, July 2003.

[RFC4242] Venaas, S., Chown, T., and B. Volz, "Information Refresh Time Option for Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 4242, November 2005.

[RFC4361] Lemon, T. and B. Sommerfeld, "Node-specific Client Identifiers for Dynamic Host Configuration Protocol Version Four (DHCPv4)", RFC 4361, February 2006.

[RFC7283] Cui, Y., Sun, Q., and T. Lemon, "Handling Unknown DHCPv6 Messages", RFC 7283, July 2014.

15.2. Informative References

[LW4OVER6] Cui, Y., Sun, Q., Boucadair, M., Tsou, T., Lee, Y., and I. Farrer, "Lightweight 4over6: An Extension to the DS-Lite Architecture", Work in Progress, June 2014.

[RFC951] Croft, B. and J. Gilmore, "Bootstrap Protocol", RFC 951, September 1985.

Authors' Addresses

Qi Sun
Tsinghua University
Beijing 100084
P.R. China

Phone: +86-10-6278-5822
EMail: sunqi@csnet1.cs.tsinghua.edu.cn

Yong Cui
Tsinghua University
Beijing 100084
P.R. China

Phone: +86-10-6260-3059
EMail: yong@csnet1.cs.tsinghua.edu.cn

Marcin Siodelski
Internet Systems Consortium
950 Charter Street
Redwood City, CA 94063
USA

EMail: msiodelski@gmail.com

Suresh Krishnan
Ericsson
8400 Blvd. Decarie
Town of Mount Royal, Quebec
Canada

EMail: suresh.krishnan@ericsson.com

Ian Farrer
Deutsche Telekom AG
CTO-ATI, Landgrabenweg 151
Bonn, NRW 53227
Germany

EMail: ian.farrer@telekom.de